

REMARKS

This Amendment is submitted in response to the Office Action of October 20, 2005 (hereinafter "the Office Action"). Upon entry of this Amendment, claims 30 and 31 will be amended. Therefore, claims 8 and 21-31 remain pending.

In this Amendment, all references to the claims, except as noted, will be made with reference to the claim list above beginning on page 2. All references to "the Office Action," except as noted, will be referencing the most recent Office Action dated October 20, 2005. If a source document does not include line numberings, then referenced line numbers will count every printed line, except the page header, but including section headings. If there is any confusion or questions regarding any aspect of this Amendment, the Examiner is invited to immediately contact the undersigned.

Interview

The Examiner is thanked for the courtesy of the Examiner Interview given on December 7, 2005. The Examiner Interview was focused mainly on enablement issues in the objection to the specification and 35 U.S.C. § 112, second paragraph rejections, which are addressed in this Amendment. Also discussed was the reference Peterson et al. Although no agreement was explicitly reached, the interview was very helpful to Applicants in identifying key issues to be addressed and helped Applicants better understand the Examiner's positions with respect thereto.

Amendment

Claims 30 and 31 are amended to improve clarity. No new matter is entered by this Amendment.

Objection to the Specification

The specification is under objection for failing to provide proper antecedent basis for the claimed subject matter. There appear to be two separate grounds for objecting to the specification: One is what is meant by "binding" and whether it is enabled by the specification. A second grounds for objection appears to be whether and how a particular test case is automatically bound to an assertion.

1. “Binding” is adequately supported.

The Office Action asserts that the specification does not contain a detailed description of the manner and process of creating “a code segment that binds” (as set forth in claim 1) such that a skilled artisan would be able to make and use the claimed invention (Office Action, page 2, lines 15-18). The Office Action further states, “The above paragraph [referring to paragraph 32] states that ‘binding’ is done but does not even vaguely mention the process by which such binding occurs. It is totally unclear how a []sentence which uses particular keywords or phrases such as ‘required to, should or should not’ is bound to one of a plurality of tests” (Office Action, page 3, lines 5-8, which also identifies paragraph 73 of the disclosure as being relevant in a footnote).

These statements appear to raise a question as to (1) what is meant by binding and (2) whether “binding” is adequately described in accordance with 35 U.S.C. § 112, first paragraph such that a person of ordinary skill can make and use it.

The specification as originally filed sets forth, in the second full paragraph of page 12 (corresponding to paragraph 33 in the published application) states, “the embodiments of the present invention keep binding information, which is the correspondence between a test case and an assertion.” Thus, the specification establishes that a “binding” can be a correspondence between the test case and the assertion. Furthermore, the term, “binding” is a term of art in the field of computer science. As defined, for example, in Wikipedia, “In computer science, *binding* refers to the creation of a simple reference to something which is larger and more complicated and used frequently. The simple reference can be used instead of having to repeat the larger thing.”* Since the term, “binding” is consistently used throughout the specification to refer to a correspondence between a first element and a second element, and because the term, “binding” is a term of art that would have been understood by those having ordinary skill in the art, Applicants respectfully submit that the term, “binding” is well defined within the context of the present Application.

With regard to having an enabling disclosure, Applicants respectfully point out that the formation of a reference from one object to another, such as between an assertion and a test case, would be well within the skill of the ordinary computer programmer. In object oriented programming, for example, a pointer can be provided to identify the referenced object.

* [http://en.wikipedia.org/wiki/Binding_\(computer_science\)](http://en.wikipedia.org/wiki/Binding_(computer_science)). Consistent definitions within the computer industry can be found by entering “define:binding” in a Google® search.

2. Automated binding not required by specification

A separate enablement issue is raised in the Office Action at page 3, lines 8-10, wherein the Office Action asserts that “The specification does not describe how the semantic meaning of the specification assertion is obtained nor how the gist of one of the plurality of tests is deduced such that the assertion can be bound to the correct test selected from a plurality of tests.” It is true that the present Application does not identify any automated means for identifying which assertion should be bound to a particular test case. As it was originally drafted, the Application *does* identify an automated mechanism to extract assertions from an input specification. This mechanism is described in the written description, on page 23, line 21 to page 24, line 10 (corresponding to paragraph 73 of the published application) and elsewhere. Once a list of assertions are generated, they are bound to test cases as described in the Background section, page 3, line 17 to page 4, line 13 (corresponding to paragraphs 9-11 of the published application). In particular, the specification states that “Test developers . . . generate at least one test case for each testable assertion that appears in the API specification” (Application, page 4, lines 11-13; or paragraph 11, lines 2-4 of the published application). Because a patent application generally focuses on what is new, and not what is already known, the process of assigning a particular test case to an assertion is not further described. Thus, as would have been understood by a person having ordinary skill in the art upon review of the present disclosure, the software application described in the present Application binds a test case to a test assertion to automatically track the progress of the compatibility testing, but relies on the user (or “test developer”) to initially identify which assertion (or assertions) is proved by each test case, e.g., when each test case is developed.

As for the automated extraction of assertions from the input specification, Applicants respectfully point out that the syntactical analysis described will enable a person having ordinary skill in the art to extract sentences having “should,” “shall,” etc., from an input specification. Although this extraction may not function to 100% perfection in extracting each explicit and implied assertion present in an input specification, it is not required for enablement under 35 U.S.C. § 112 that an invention operate with 100% effectiveness.

Since the disclosure does not require that a test case be automatically identified from each assertion, Applicants respectfully submit that no disclosure describing how this would be done is necessary to meet the 35 U.S.C. § 112 enablement requirement.

For the reasons discussed above, Applicants respectfully request withdrawal of the outstanding objection to the specification.

Claim Rejections - 35 U.S.C. § 112

Claims 8 and 21-31 stand rejected under 35 U.S.C. § 112, second paragraph as being indefinite. Specifically, the Office Action suggests that the term, “binding,” set forth in claims 8 and 25, is not adequately defined. Applicants respectfully disagree. As mentioned above, the second full paragraph of page 12 (corresponding to paragraph 33 in the published application) states, “the embodiments of the present invention keep binding information, which is the correspondence between a test case and an assertion.” Furthermore, the term, “binding” is a term of art that would have been understood by those skilled in the art at the time the invention was made. Applicants respectfully submit that the definition presented in the Office Action (page 4, lines 2-5) of “dynamic binding,” which is a type of binding, from the Microsoft Computer Dictionary, is not relevant for the general term “binding” as it is used in the present Application, although it is noted that the provided definition does refer to a correspondence between one object and another (specifically, a software routine being bound to a data object).

In addition to the above, the Office Action suggests that the phrase, “top-level, package-level, class-level and constructor/method/field” is indefinite. Applicants have amended claims 30 and 31, which contain this phrase, to be more clear. Applicants point out that the terms, “package” “class” “constructor” “method” and “field” are terms of art that individuals skilled in the art of API development would be familiar with. Applicants have amended the claims to more clearly identify the metes and bounds of the claims in terms of alternative versus inclusive terminology. The Office Action questions what a top-level specification is (page 4, lines 14-15). Applicants respectfully submit that, as any person having ordinary skill in the art would understand, a top-level specification is a specification that applies to the entire API (or other software project) being developed. A particular software project such as an API may comprises multiple packages, each of which may be a subdivision of the project. Other elements mentioned may make up a package.

Claim Rejections - 35 U.S.C. § 103

Claims 8 and 21-29 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent 5,805,795 issued to Whitten (“Whitten”) in view of U.S. Patent 5,892,947

issued to DeLong et al. ("DeLong"), and further in view of U.S. Patent 6,598,015 issued to Peterson et al. ("Peterson"). Applicants respectfully traverse because the prior art fails to teach or suggest each and every limitation set forth in the claims, and because there was no suggestion or motivation in the prior art to combine and/or modify the references as proposed in the Office Action.

1. The prior art does not teach binding a test case with an assertion

The Office Action identifies Whitten as disclosing a code segment that binds each testable assertion to one of a plurality of tests that test the testable assertion (Office Action, page 5, lines 21-23). Applicants respectfully disagree. Whitten discloses a system for selecting optimum test cases for fully testing a software product in the shortest amount of time (col. 5, line 66 to col. 6, line 6). Whitten provides in Figure 2, a list of test cases 10 that identifies which program blocks are tested by the test case. That is to say, which sections of a program's code are executed by the test case. By identifying which sections of code are executed for each test case, a selection of test cases can be made that exercise all the code blocks in the fewest amount of time. An assertion is not a code block, but a human-language description of proper program behavior (col. 1, lines 29-33). Since an assertion is not a code block, Whitten does not provide any suggestion of a binding between a test case and an assertion. None of the other references teach or suggest binding a test case with an assertion. Since none of the references teach or suggest binding a test case with an assertion, Applicants respectfully submit that independent claims 8 and 25 are not made obvious by the prior art of record and should therefore be allowed. Claims 21-24 and 26-31 depend from one of claims 8 or 25 and should therefore be allowed for at least the same reasons.

2. The prior art does not teach associating a context with each testable assertion

The Office Action contains an admission that Whitten does not disclose "a code segment that associates a context with each of the testable assertions . . ." (page 6, lines 19-20). However, the Office Action states that "Peterson discloses maintaining the context provided by the formatting code nodes within a tree structure that is extracted from a document to be translated" (page 7, lines 2-3). Applicants respectfully assert that this is not the same thing as creating an association between a context and a testable assertion. Even assuming, *arguendo*, that the Office Action correctly characterizes Peterson, in suggesting that Peterson maintains a context, this does not suggest forming an association between a context and an assertion. For obviousness under 35 U.S.C. § 103(a), each and every limitation must

be taught or suggested by the prior art reference, or references when combined or modified (MPEP 2143). Even if it is true that Peterson maintains a context, there is no suggestion in any reference of record to create an association between a context and an assertion.

Since the prior art lacks any suggestion to create an association between a context and an assertion, Applicants respectfully submit that independent claims 8 and 25 are patentable over the prior art of record. Furthermore, remaining pending claims 21-24 and 26-31 depend from one of claims 8 and 25 and should therefore be allowed for at least the same reasons as claims 8 and 25.

3. The prior art lacks requisite motivation to combine and/or modify the references

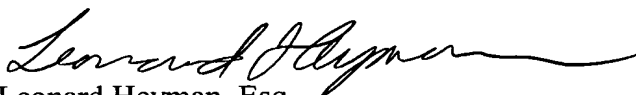
The Office Action states that “It would have been obvious . . . to modify Whitten to include ‘a code segment that associates a context with each of the testable assertions . . . [’] based on the disclosure of Peterson for the purpose of making use of a previously translated document [col. 2, line 39-40]” (page 7, lines 3-10). For an obviousness type rejection, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or combine reference teachings. See MPEP 2143.01. Peterson discloses a system to assist document translators (for translating an document stored in electronic form from one human-spoken language to another) by performing pattern matching to leverage existing previously translated documents for display along with a document to be translated and a text window for entering translation text (col. 5, line 52 to col. 6, line 12). Applicants do not understand how a desire to make use of a previously translated document would motivate a person to combine the teaching of Peterson and Whitten. Even if an input specification were translated from a different language, there is no guidance in the prior art as to how the systems of Peterson and Whitten can be combined to create a working system. There is absolutely no recognition in the prior art of any advantage to be gained by associating the assertions with contexts. Without any recognition in the prior art of an advantage, there was no motivation.

Since the prior art lacked motivation to combine the references as proposed in the Office Action, Applicants respectfully submit that claims 8 and 21-29 are allowable over the prior art of record. Furthermore, since claims 30 and 31 depend from one of claims 8 and 25, Applicants respectfully submit that claims 30 and 31 should be allowed for at least the same reasons as claims 8 and 25.

For the reasons discussed above Applicants respectfully submit that the present Application is now in condition for Allowance. A Notice of Allowance is therefore respectfully requested.

If the Examiner has any questions concerning the present amendment, the Examiner is kindly requested to contact the undersigned at (408) 774-6933. If any other fees are due in connection with filing this amendment, the Commissioner is also authorized to charge Deposit Account No. 50-0805 (Order No. SUNMP013). A duplicate copy of the transmittal is enclosed for this purpose.

Respectfully submitted,
MARTINE PENILLA & GENCARELLA, LLP


Leonard Heyman, Esq.
Reg. No. 40, 418

710 Lakeway Drive, Suite 200
Sunnyvale, CA 94085
Telephone: (408) 749-6900
Facsimile: (408) 749-6901
Customer Number 32291